FAST SPEAKER CHANGE DETECTION FOR BROADCAST NEWS TRANSCRIPTION AND INDEXING

Daben Liu *Francis Kubala*

BBN Technologies, GTE Corporation

70 Fawcett Street, Cambridge, MA 02138, USA

ABSTRACT

In this paper, we describe a new speaker change detection algorithm designed for fast transcription and audio indexing of spoken broadcast news. We have designed a two-stage algorithm that begins with a gender-independent phone-class recognition pass. We collapse the phoneme inventory to only 4 broad classes and include 4 different models for non-speech, resulting in a small fast decoder that runs in less than 0.1 times real-time. The second stage of the SCD algorithm hypothesizes a speaker change boundary between every phone in the labeled input. The phone level time resolution in our approach permits the algorithm to run quickly while maintaining the same accuracy as a frame level approach. Applying the new algorithms to a large sample of broadcast news programs resulted in improvements in speaker change detection accuracy, speech recognition accuracy, and speed.

Keywords: speaker change detection, automatic transcription, audio indexing, evaluation metrics

1 INTRODUCTION

In many applications of speech recognition and speaker identification, it is very often the case that the speech is from one or two speakers and the speech is given to the system in discrete utterances. In a more complicated domain, such as broadcast news transcription, the speech can come from an unknown and widely varying number of speakers and there are no breaks between utterances. In such a domain, Speaker Change Detection (SCD) is needed.

SCD breaks up the continuous input into discrete utterances that are easy to process in large vocabulary speech recognizers. More importantly, it provides the recognizer with input that is homogeneous in speaker so that speaker normalization techniques can be used more effectively. SCD also forms the basis for speaker clustering that is used for speaker adaptation. In an audio indexing system, SCD provides a structural summary of the speaker turns contained in a conversation. Speaker changes are often cues for boundaries between programs, topics, or scenes in a multi-media application.

Speaker change detection in continuous audio streams of broadcast news is a difficult task due to the highly variable noise and channel conditions and the fact that the true number of speakers is unknown. Previous work in SCD in the context of speech recognition includes [1] [2] [4] [5] [6]. Several distance measures have been used to calculate the speaker differences, such as, Generalized Likelihood Ratio (GLR) [2], Kullback-Leibler distance (KL) [4], Bayesian Information Criterion (BIC) [6].

We have designed a two-stage algorithm for SCD. It begins with a gender-independent phone-class recognition pass to detect and classify non-speech into silence, music, noise, or other non-speech intervals. It also locates boundaries between 3 broad phoneme classes. The second stage of the SCD algorithm hypothesizes a speaker change boundary between every phone in the labeled input. Experiments with the new algorithm have shown very promising results.

One of the problems in SCD area is the lack of a common metric to evaluate different systems on the same problem. So we begin by defining the performance metric that we use to measure SCD accuracy. After that, we describe our new algorithm in detail. In Section 6, we present SCD results on broadcast news test data and compare our results with other work in the area.

2 EVALUATION METRICS

We treat SCD as a detection problem and evaluate SCD algorithms in terms of the two types of errors that can occur. One is False Acceptance (Type I) error, in which a putative boundary is not a true boundary. The other is False Rejection (Type II) error, in which a true boundary is not detected. Typically, manually produced segmentations are used to define ground truth. One problem with this approach is that different annotators are quite likely to pick different points. As a result, inter-annotator variability is conflated with the error measure. We propose a procedure that avoids this problem.

Typically, a speaker change occurs in one of two forms:

- There is a short period of silence or other non-speech between two speakers. In broadcast news, there are quite a few non-speech events other than silence, such as music, laughter, breath, lip-smack, coughing, etc. All of them may happen at the speaker boundaries. In this case, any detected changes within this non-speech period should be considered as correct. Only in the extreme situation where the duration of the period is zero (no gap), does the correct region degrade to a single point.
- The speech of the two speakers overlaps. In this case, the overlapped region can be considered as the correct region and any detection within this region is correct.

As we can see, correct changes can usually be represented as regions rather than single points. Denoting all the reference changes as $[a_i, b_i]$ where $i = 1,..., N$ and N is the number of true changes, the two types of errors can be defined as follows:

False acceptance error occurs if the hypothesized change does not fall into any of the intervals: $[a_i - \alpha, b_i + \alpha]$.

False rejection error occurs if for some j, there is no detected change within the interval: $[a_j - \alpha, b_j + \alpha]$.

α is a tolerance factor that can be set according to different requirement. We select α to be 100ms, which is about the average length of one phoneme. This is a very conservative tolerance.

To generate this ground truth reference we align the reference transcription with the acoustic data with a constrained Viterbi decoding. By doing this, non-speech regions will be labeled. We then map each hand-labeled speaker boundary to the enclosing non-speech regions. Where no non-speech regions are found, the manual boundaries are retained.

3 PHONE-CLASS DECODE

Though most of the non-speech events are notoriously bad for speech recognition accuracy, they possess valuable information about speaker changes. As we have observed, more than 80% of the true speaker changes happen at non-speech. Thus we would be more confident if a change is hypothesized at non-speech region. Detecting non-speech also permits us to exclude non-speech frames when clustering or identifying speakers. Only frames of data containing speech are useful in determining speaker differences. Furthermore, if the SCD is to be used as a front-end for speech recognition, it is important that speaker boundaries are not hypothesized in the middle of words.

Because of the variety of non-speech events, and the low SNR of many of them, energy based methods are not effective for detecting non-speech. Other more sophisticated methods have been used to detect non-speech. We prefer to employ a phoneme decode for this purpose. The BBN phoneme decoder as described in [1], is part of the state-of-the-art BBN BYBLOS Broadcast News Transcription System. The approach described here will be designated as the baseline SCD system for the remainder of this paper.

We create a gender-dependent and context-independent phone HMM. 45 context-independent phone models are trained for each gender. A silence model is trained with samples of silence, music and noise. These 91 phone models are then used to decode the speech. The output is a sequence of phones with gender and silence labels. This approach works very well in detecting silence and music. Also with gender changes labeled, speaker changes between different genders are easily obtained. However, there are some drawbacks. There are some cases in which a sequences of gender errors are made on short segments in particular. Background noise may also affect the gender detection. This makes the output gender labels somewhat noisy and some complicated heuristic rules are needed to smooth the results before they can be useful. The 91-phone baseline decoder is also quite slow.

Since the goal is to detect non-speech, there is no need to explicitly distinguish between phones. From speech production theory, phones can be roughly classified into 4 groups: vowels, nasals, fricatives or sibilants, and obstruents. Within each group, the phones have similar acoustic characteristics. Vowels and Nasals are similar in that they both have pitch and high energy. They both are quasi-periodic. We therefore put them together and build 3 phone classes, as shown in Table 1.

Vowels and Nasals	AX, IX, AH, EH, IH, OH, UH, EY, IY, AY, OY, AW, OW, UW, AO, AA, AE, EI, ER, AXR, M, N, NX, L, R, W, Y
Fricatives	V, F, HH, TH, DH, Z, ZH, S, SH
Obstruents	B, D, G, P, T, K, DX, JH, CH

Table 1. Phone classes used for SCD.

To classify the non-speech events, 4 new acoustic models are created for music, laughter, breath, and lip-smack, respectively. Together with silence, we have 8 phone classes. By doing this, the number of active nodes during decode is largely reduced and the speedup is significant.

We use a 5-state HMM to model each of the phone classes. One codebook with 64 diagonal Gaussian Mixture Models (GMM) is shared by the 5 states from the same phone class and for each state a Gaussian mixture weight is trained. The scheme is called Phonetically Tied Mixture (PTM) modeling [1]. Twenty hours of speech from broadcast news training corpus are used to train the 8 phone-class models.

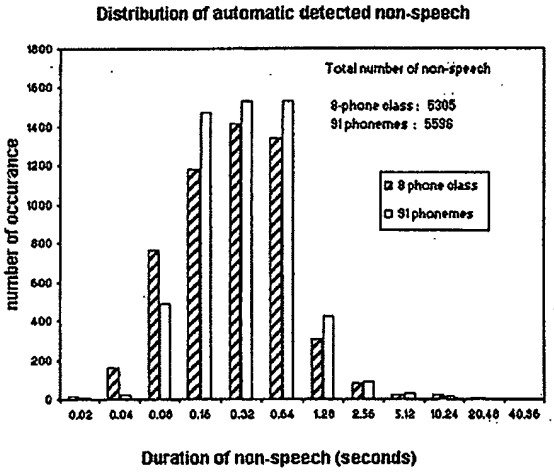

Figure 1. Distribution of automatically detected non-speech

Decoding with these phone-class models produces a sequence of phone classes and non-speech labels with time. Figure 1 shows the distribution of non-speech detected on a 3-hour broadcast news with both 8-phone and 91-phone models. Note that the majority of the non-speech is silence. The 8-phone model finds fewer medium length non-speech segments but more short ones (less than 0.1 seconds). The long ones (longer than 5 seconds) are mostly music or gaps between programs, which both have detected successfully. The overall distributions are quite similar between these two. The total number of segments detected is also very close.

Note that we use a gender-independent approach in the phone-class decode. We believe gender difference would be easily detected with speaker change detection where not only the gender features but also other speaker features are utilized to detect the difference. Doing so we can also avoid using any complicated heuristic rules that may not be robust.

4 SPEAKER CHANGE DETECTION

In the BBN Byblos baseline system where 91-phone decode is used, speaker clustering is applied to separate speakers [1][3]. Speech is chopped on silence and gender changes to produce uniform-length segments. Hierarchical clustering is implemented to group segments into clusters for unsupervised adaptation. Based on the metrics we proposed in section 2, the speaker change errors are quite high (see section 5). This is because the clustering approach does not attempt to find the true changes of speakers.

In this section, we describe a speaker change detection algorithm, which utilizes the label information from the phone/non-speech sequence produced by the phone-class decode.

- The distance measure criterion

Given two sets of data $\mathbf{x} = \{x_i, i = 1...N\}$ and $\mathbf{y} = \{y_j, j = 1...M\}$ where x_i and y_j are the cepstral vectors. We wish to test the hypothesis:

H_0: x and y are produced by the same speaker

H_1: x and y are produced by different speakers

Assuming that both x and y are from independent Gaussian processes, the generalized likelihood ratio test [2] would be:

$$\lambda = \frac{L(\mathbf{z}; \mu_z, \Sigma_z)}{L(\mathbf{x}; \mu_x, \Sigma_x) L(\mathbf{y}; \mu_y, \Sigma_y)}$$

where $L(\mathbf{v}; \mu_v, \Sigma_v)$ is the maximum likelihood of v, and z is the union of x and y.

It is usually the case that the more data we have for estimating the Gaussians, the higher the λ is [2]. To alleviate this bias, a penalty factor is added such that the test we are using changes to:

$$\lambda' = \frac{\lambda}{(N+M)^{\theta}}$$

θ is determined empirically. This penalized likelihood ratio is similar to BIC used by [6]. Experiments have shown that it is more robust than the plain likelihood ratio.

- **The critical region**

For a standard hypothesis test, the critical region would be that:

If $\lambda' \le \lambda_{th}$, then H_0 holds

Otherwise, H_1 holds

λ_{th} is the threshold to be set based the distribution of H_0 and H_1.

In our case, however, we have extra information to assist the assessment. As we mentioned before, a true change is more likely to happen at non-speech region. Also cutting on a non-speech is less damaging than cutting in the middle of a word. Based on this assumption, we actually uses a higher threshold for changes that are not on non-speech. So the critical region becomes:

Test at non-speech region:

<using the standard test>

Test at speech region:

if $\lambda' \le \lambda_{th} + \alpha$, then H_0 holds

Otherwise, H_1 holds

In here, $\alpha \ge 0$.

- **Phone-based speaker change detection**

We implemented a sequential procedure, which increments one phone at a time and hypothesizes speaker changes on each phone boundary. The sequence can be described with the flowchart in Fig 2.

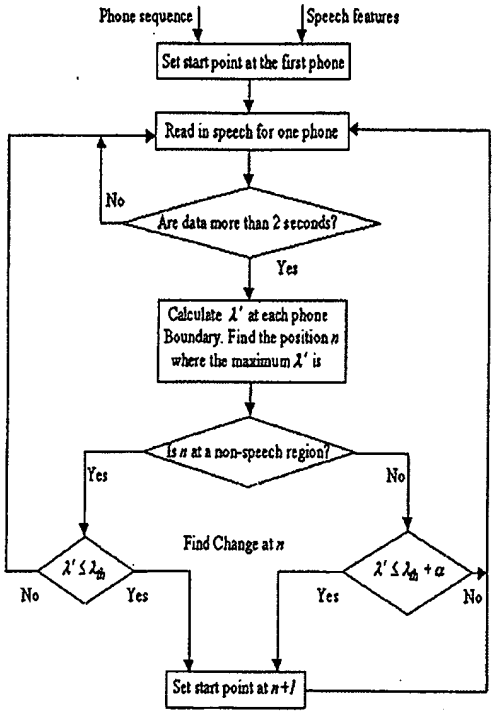

Figure 2. Speaker Change Detection Flow-chart.

In the implementation, the thresholds are set such that the total errors (false acceptance and false rejection) are a minimum. The procedure is nearly causal. It looks ahead only 2 seconds in order to get enough data for the detection. The phone level time resolution in our approach permits the algorithm to run very quickly while maintaining the same accuracy as a frame level approach.

5 EXPERIMENTS AND RESULTS

We implemented our algorithm on Hub4 1997 evaluation data, which is provided by National Institute of Standards and Technology (NIST). It contains about 3 hours of broadcast news. NIST has also provided hand-generated segments based on speakers and speaking conditions, including prepared speech, spontaneous speech, telephone speech, speech with music background and speech with background noise. There are totally 620 speaker/condition-based segments. If only counting the speaker changes, there are 482 true speaker turns.

To build the reference for evaluating SCD, forced alignment is implemented on each speaker boundary to find possible non-speech. In the case when there is a long music piece (> 2 seconds) between two speakers, we recognize it as a true speaker change.

For comparison, we use the segmentation from BBN Hub4 1997 evaluation system as the baseline, which uses 91-phone decode for silence detection and speaker clustering for speaker separation. The new SCD algorithm is evaluated with respect to our SCD metric, effect of speech recognition word-error-rate and runtime speed.

- **SCD performance**

Comparative results are shown in Table 2 for our new SCD algorithm compared to the BBN baseline system and a

segmentation produced by CMU in the 1997 Hub4 evaluation [4].

	segments detected	False rejection	False acceptance	speakers/ segment
BBN	515	49.2%	56.3%	1.250
CMU	769	42.8%	64.1%	1.239
HTK	749	N/A	N/A	1.173
New SCD (all data)	483	30.0%	25.0%	1.122
New SCD (speech only)	475	29.5%	20.0%	1.120

Table 2. Comparative SCD performance.

They are also compared to HTK segmentation [5] by average number of speakers per segment, the measure that is used by HTK. The problem with this measure is that it is partially dependent on the number of segments hypothesized. IBM has also reported their segmentation performance in [6]. However, it does not use speaker changes as the reference and it is not clear how the errors are defined. As a result, it is hard to compare with.

The false rejection and false acceptance results shown in Table 2 assume a tolerance factor of 100ms. The new SCD algorithm is obviously superior to our baseline system in detecting speaker changes. We can also see that non-speech data hurts the false acceptance performance. In the experiments, we observed that when a long silence is present in the middle of a speaker turn, it is likely to be detected as a change if the silence is used to calculate the distance.

- **Word-Error-Rate (WER)**

Applying the new SCD in the BBN BYBLOS transcription system, we see that WER is decreased by higher accuracy SCD.

	WER without adaptation
BBN baseline	22.7%
With new SCD	21.4%
With true speaker turns	20.7%

Table 3. Comparison of Speech Recognition Word Error Rate (WER) as a function of SCD accuracy.

We also provide the WER result with true speaker turns. BBN baseline system has a WER that is 2% absolute higher than that using the true segments. After using speaker change detection, the difference to truth is only 0.7%.

We attribute this improvement to the fact that SCD provides more homogeneous segments in term of speaker. The cepstrum normalization can be applied to better speaker turns and thus can be more effective. Also precise cuts on speaker change give correct sentence beginning, which may also help language model alignment.

- **Speed**

Because of the introduction of phone-class, the phone decode has been sped up from 10 times of real time to 0.07 times of real time, which is shown in Table 4. The overall speed of the new SCD algorithm has been sped up by a factor of 30 compared to the baseline.

	Throughput on a Pentium II 450MHz processor (Real Time)		
	Phone decode	SCD	Total throughput
BBN baseline	10.00x	0.30x	10.30x
New SCD	0.07x	0.28x	0.35x

Table 4. Improvement in speed with the new SCD algorithm.

6 CONCLUSION

In this paper, we present a new algorithm for speaker change detection in continuous speech with multiple speakers and varying environment, such as broadcast news. The results are promising and the speed of the procedure is less than real-time. A new evaluation metric for SCD is proposed such that valuable research work from different research sites can be comparable. We have also shown that speaker change detection improves speech recognition accuracy by making speaker normalization more effective.

ACKNOWLEDGEMENT

This work was supported by the Defense Advanced Research Projects Agency and monitored by the Air Force Research Laboratory under contract No. F30602-97-C-0253. The views and findings contained in this material are those of the authors and do not necessarily reflect the position or policy of the Government and no official endorsement should be inferred.

REFERENCES

1. Kubala, F., et al., "The 1997 Byblos System Applied to Broadcast News Transcription", *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, February 1998

2. Gish, H., M. H. Siu, R. Rohlicek, "Segregation of Speakers for Speech Recognition and Speaker Identification," *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, Toronto, Canada, vol. 2, pp. 873-876, May 1991

3. Jin, H., F. Kubala, R. Schwartz, "Automatic Speaker Clustering," *Proceedings of the DARPA Speech Recognition Workshop*, pp. 108-111, February 1997

4. Siegler, M., et al, "Automatic Segmentation Classification and Clustering of Broadcast News Audio," *Proceedings of the DARPA Speech Recognition Workshop*, pp. 97-99, February 1997

5. Hain, T., et al, "Segment Generation and Clustering in the HTK Broadcast News Transcription System," *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, February 1998

6. Chen, S., P. Gopalakrishnan, "Speaker, Environment, and Channel Change Detection and Clustering via the Bayesian Information Criterion," *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, February 1998